FUJITSU

14 November 2011

On Non-Blocking Collectives in 3D FFTs

Dr Radhika Saksena
Environment and Health Research Division
Fujitsu Laboratories of Europe Ltd

shaping tomorrow with you

Non-Blocking Collectives

FUJITSU

- Non-blocking collectives: Ialltoall, Ibcast, etc. Once non-blocking operation is posted, communication progresses:
  - Asynchronously or
  - Progress communication manually by calling MPI_Test()/NBC_Test()

- Not in MPI 2.0 library implementations but proposed in MPI 3.0 draft specification and widely expected:

  http://meetings.mpi-forum.org/presentations/MPI_Forum_SC10.ppt.pdf

- Use libNBC [1] to provide non-blocking collective functions. libNBC depends on non-blocking point-to-point MPI_Isend and MPI_Irecv for implementing the collectives.

Three-Dimensional FFTs

FUJITSU

FFTs are important for performance of a range of HPC applications

- Computational Quantum Chemistry – Many small independent FFTs
- Molecular Dynamics (Materials Science and Biophysics) – Small FFTs of sizes <= 512^3
- CFD (DNS, spectral methods) – Single large FFTs O(1000^3)

Distributed 3D FFTs

FUJITSU

Basic Algorithm to perform distributed 3D FFT:

1D decomposition on a processor (MPI task) grid

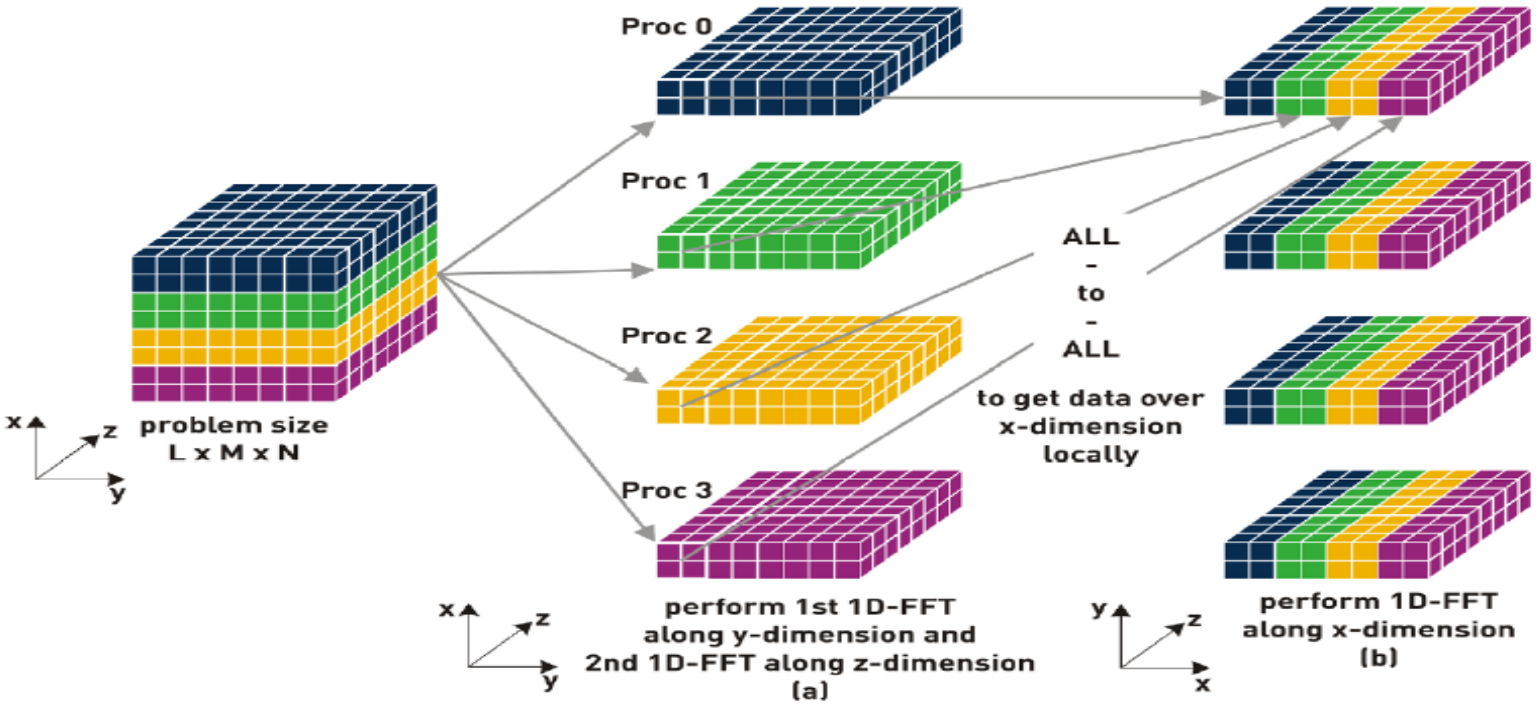


Figure 3: 1D decomposition of 3D FFT [8]

Figure from: Heiki Jagode, http://www2.epcc.ed.ac.uk/msc/dissertations/dissertations-0506/hjagode.pdf.

Maximum parallel decomposition along one axis is restricted by the array size of the longest axis

Distributed 3D FFTs

FUJITSU

2D decomposition of the MPI task grid for volumetric decomposition

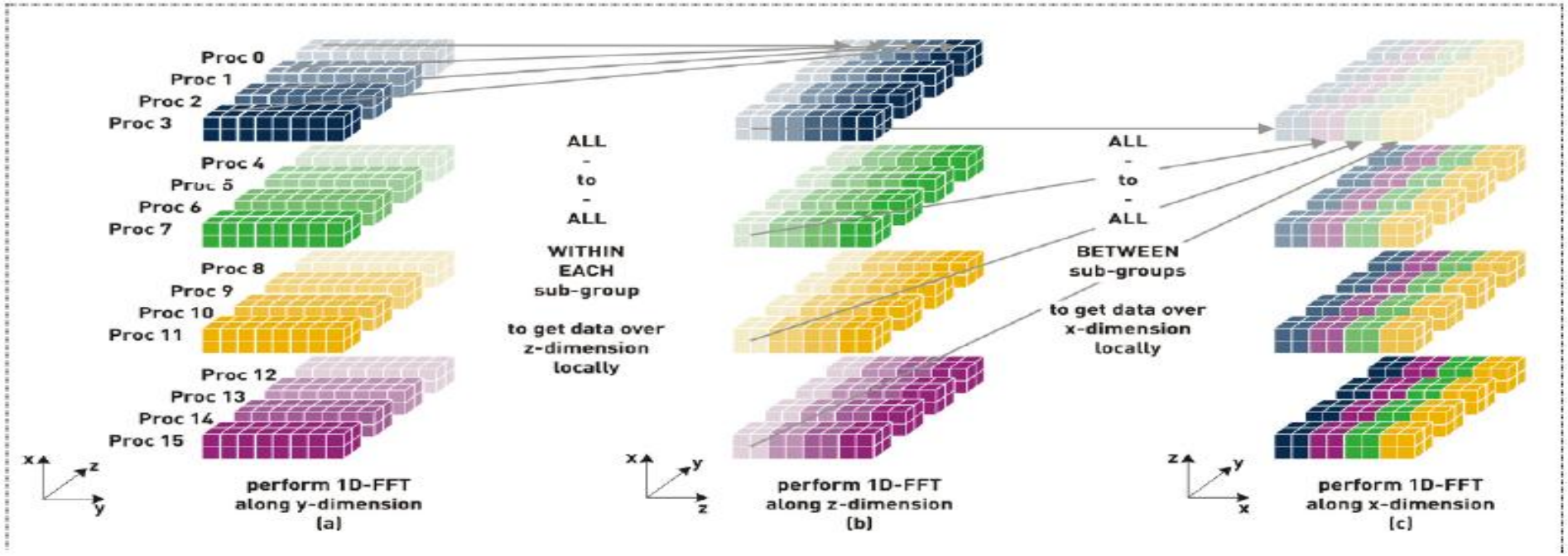


Figure 4: 2D decomposition of 3D FFT [8]. The coordinate system is rotated in each step for better clarity of the communication and datalayout.

Figure from: Heiki Jagode, http://www2.epcc.ed.ac.uk/msc/dissertations/dissertations-0506/hjagode.pdf.

Distributed 3D FFTs

FUJITSU

Scaling of parallel FFTs bound by the requirement for global all-to-all communications to perform the distributed transpose.

Already a bottleneck on petascale architectures – future of FFTs in an exascale era?

Some approaches to prolong the life of FFTs in post-petascale, multi-core era:

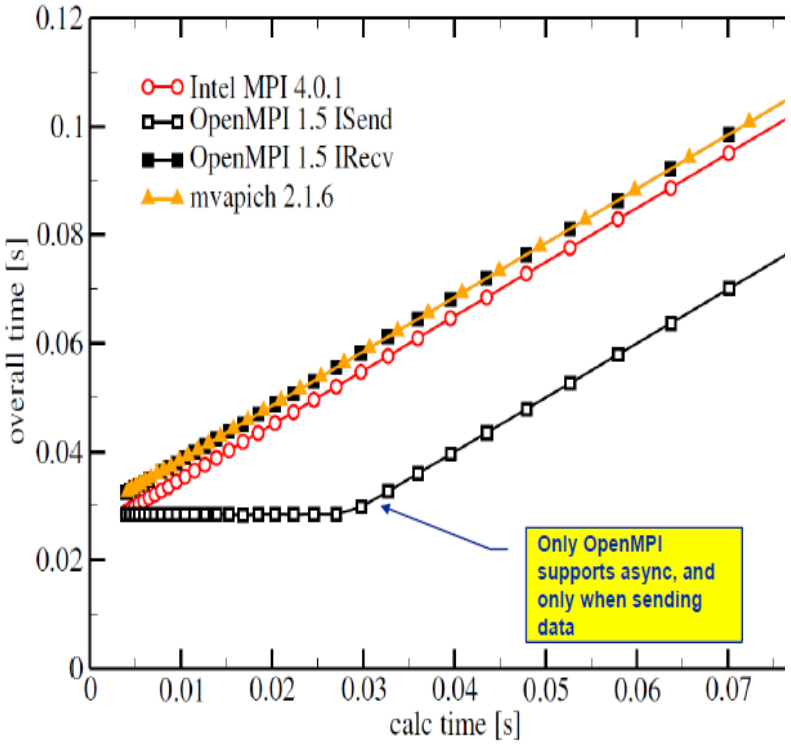
- Hybrid MPI/OpenMP or MPI/Pthreads
- Non-blocking collectives
- One-sided communication

Distributed 3D FFTs – Non-blocking Collectives

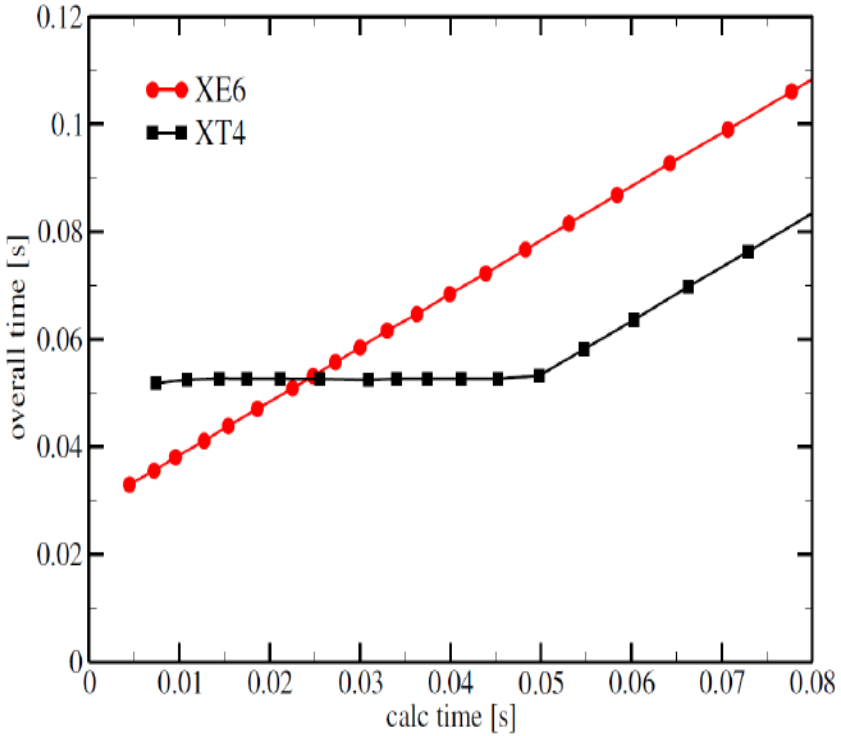
FUJITSU

Non-blocking collectives – hybridization alone will not help communication bound apps like FFTs [Hager CUG10]

- Internode results for Westmere cluster (QDR-IB)



- Internode results for Cray XT4 and XE6



Will need to manually overlap communication and computation with: 1) MPI_Test() for NB collectives in MPI 3.0? 2) Explicitly assign a thread to progress communication asynchronously

Distributed 3D FFTs – Non-blocking Collectives

FUJITSU

Non-blocking collective communication implementation by Kandalla *et al.* [5] at ISC'11 for multiple independent FFTs:

1D FFT in x for V_1
transpose x and y of V_1
1D FFT in y for V_1
Initiate transpose y and z of V_1
do $V_j = V_2$ to V_n
    1D FFT in x for V_j
    transpose x and y of V_j
    1D FFT in y for V_j
    Initiate transpose y and z of V_j
    Wait for transpose complete for V_{j-1}
    1D FFT in z for V_{j-1}
enddo
Wait for transpose complete for V_n
1D FFT in z for V_n

Fig. 3 Algorithm for the forward transform in the redesigned multi-variable, pipelined, overlapped version

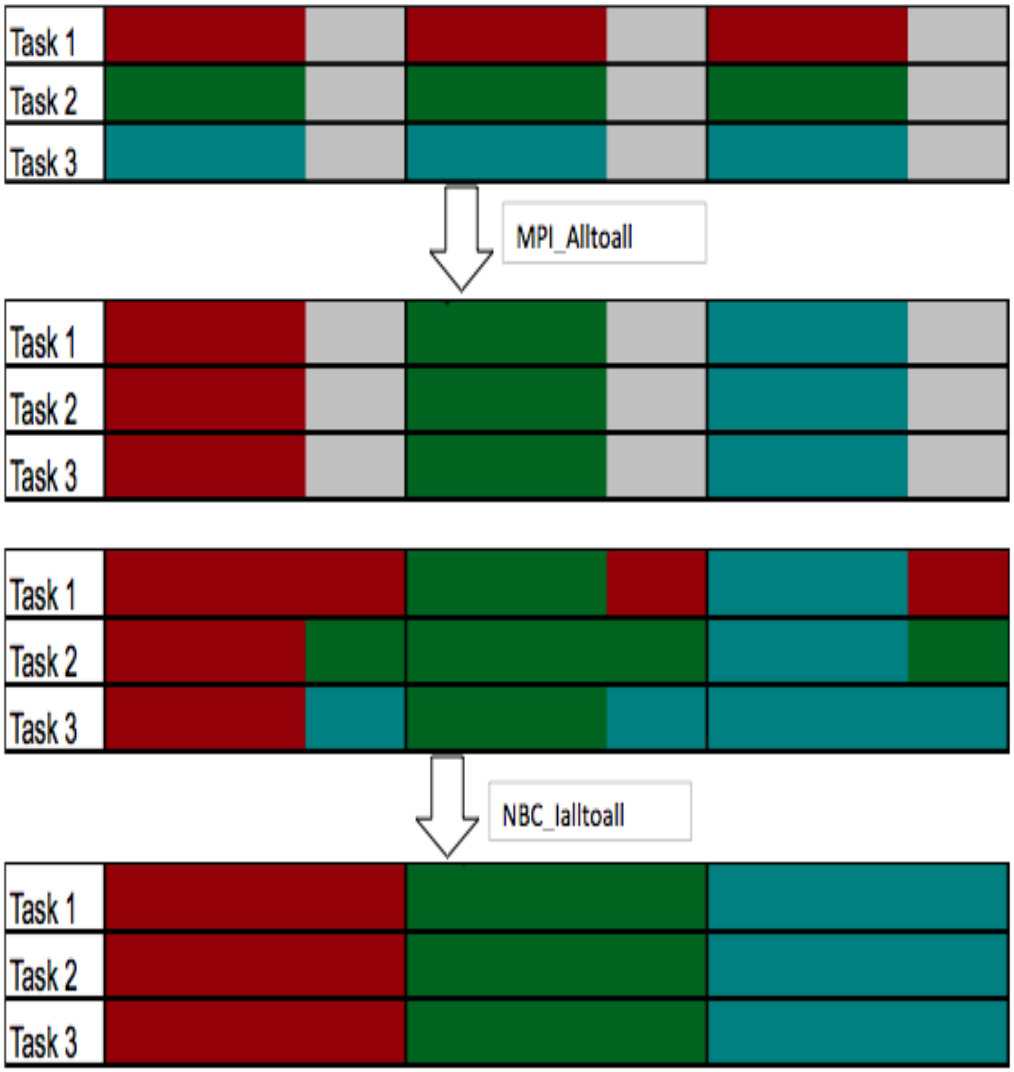
- Re-designed P3DFFT library to overlap the Alltoall operations with application-level computation
- Report host-based non-blocking Alltoall schemes faster by 6% compared to the default-blocking version, by 23% with offloading progression to hardware.

Distributed 3D FFTs – Non-blocking Collectives

FUJITSU

Our approach for non-blocking collectives in a *single 3D* FFT:

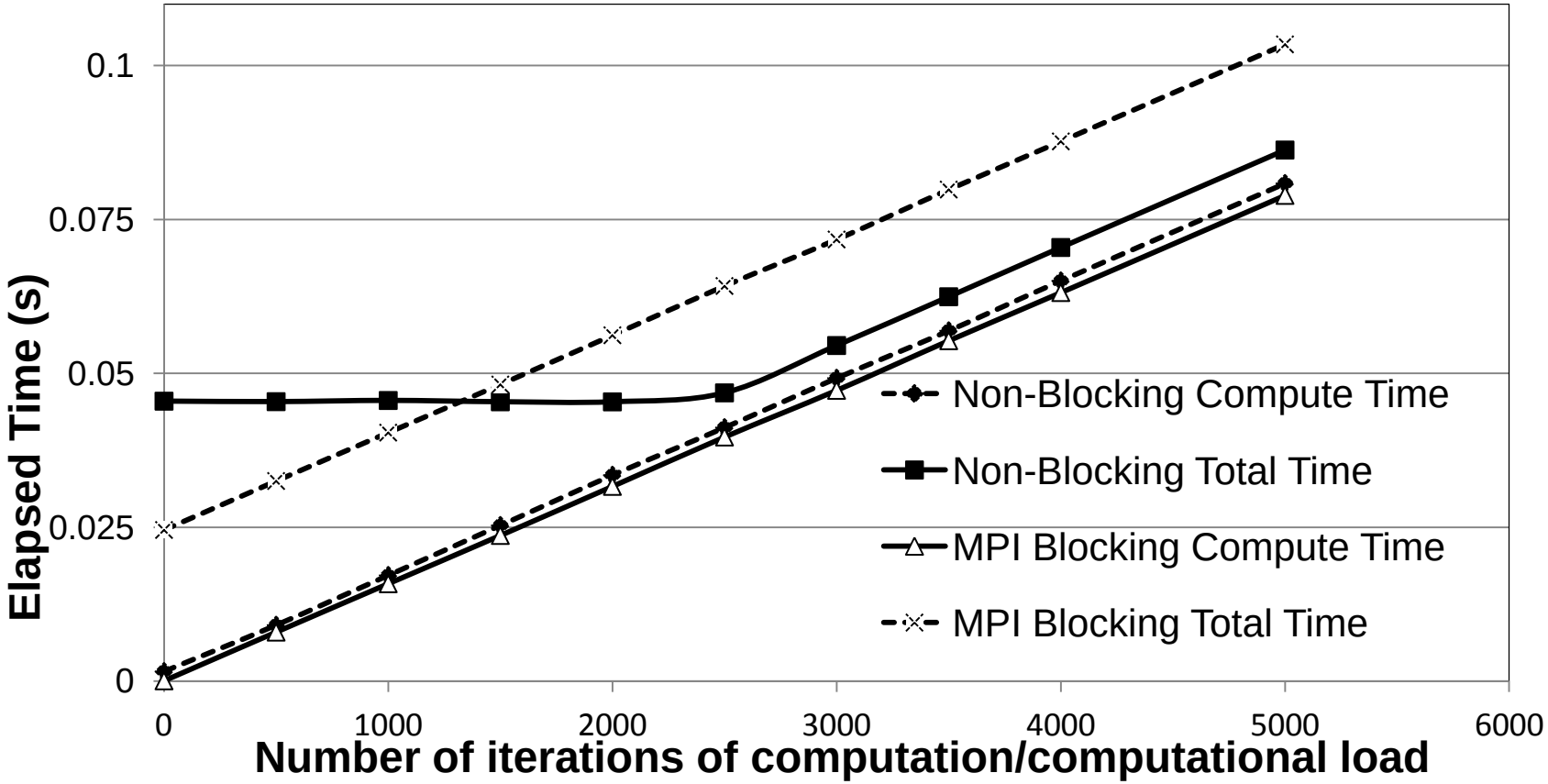
```
1D FFT in the Z dimension
1D FFT in the Y dimension
MPI_Alltoall(sendbuf,(3*size)/4, mpi_double_complex,...,
                                 recvbuf,(3*size)/4,...);
NBC_Ialltoall(sendbuf + (3*size)/4,count/4,...,
                  recvbuf + (3*size)/4,size/4,...,handle);
1D FFT in the X dimension for data of size (3*size)/4
{
call NBC_Test(handle) regularly
}
NBC_Wait(handle);
1D FFT in the X dimension for data of size (size/4)
```



Distributed 3D FFTs - NBC Collectives

FUJITSU

Implementation: Test of overlap of communication with computation with NBC_Ialltoall in libNBC:

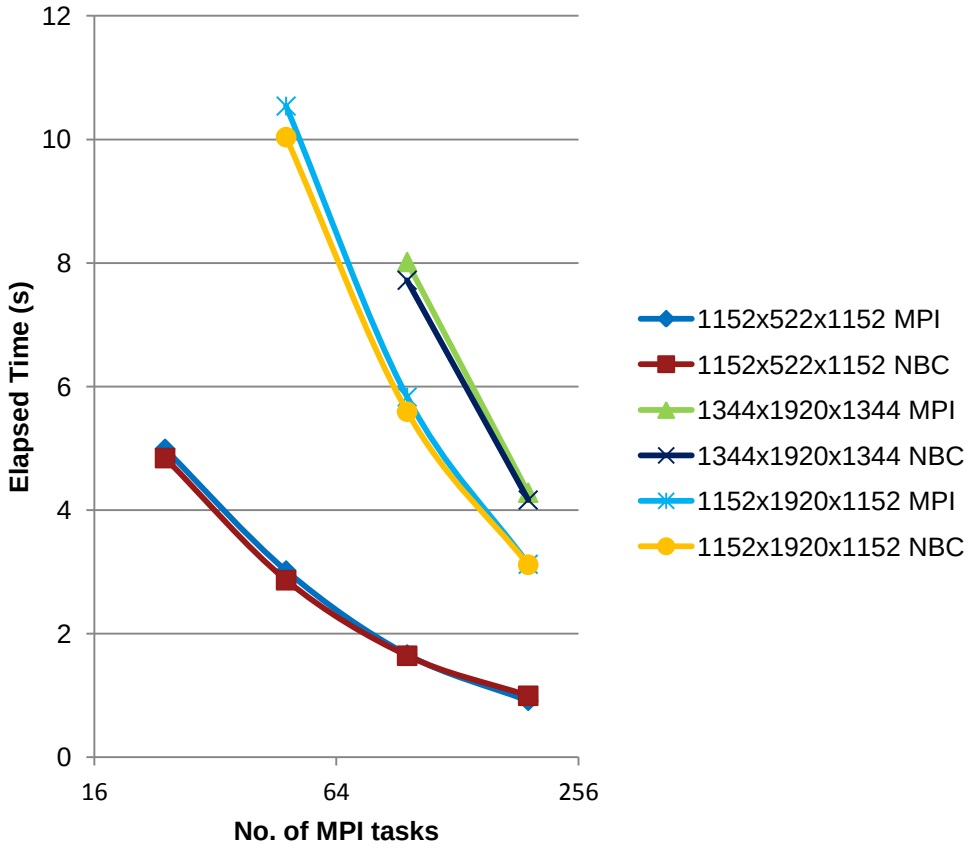


Distributed 3D FFTs - NBC Collectives

FUJITSU

Results on the BX900 Intel Westmere Cluster with OpenMPI

No. of Nodes	No. of MPI tasks	% Improvement
Input Size: 1152x522x1152 (packed)		
16	192	-8.2%
8	96	1.2%
4	48	5.2%
2	24	3.0%
Input Size: 1152x1920x1152 (packed)		
16	192	0.2%
8	96	4.3%
4	48	5.0%
Input Size: 1344x1920x1344 (packed)		
16	192	2.8%
8	96	3.7%

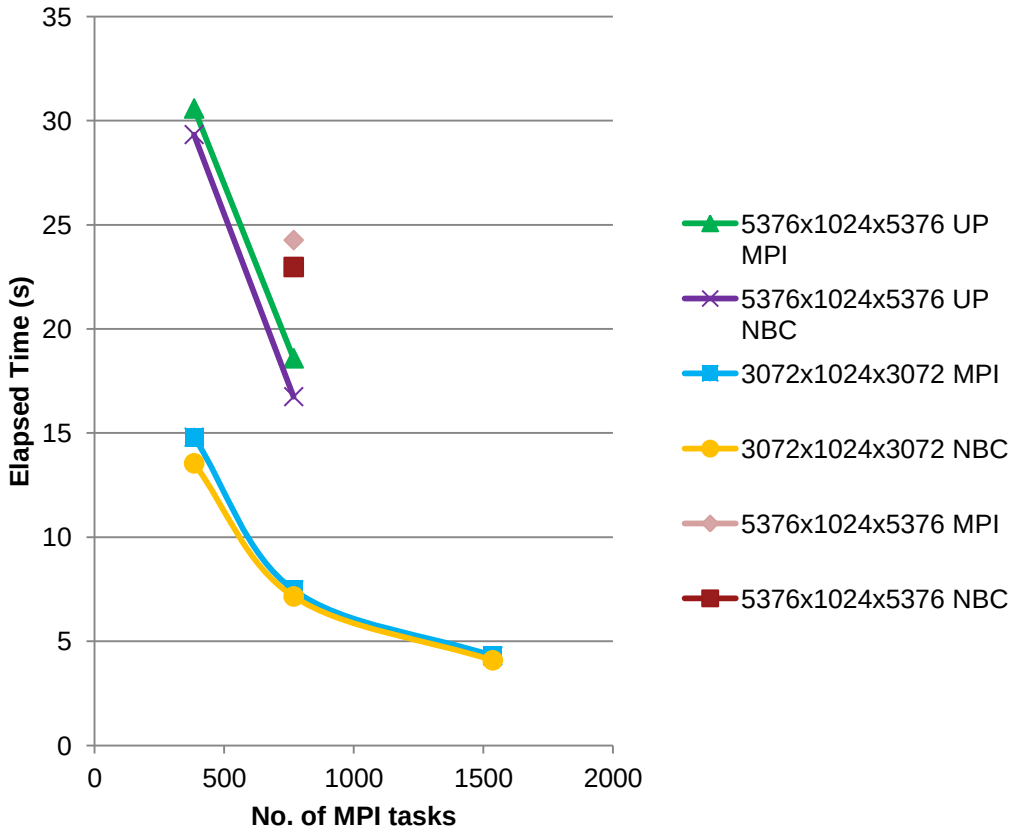


Distributed 3D FFTs - NBC Collectives

FUJITSU

Results on HECToR, Cray XE6

No. of Nodes	No. of MPI tasks	% Improvement
Input Size: 3072x1024x3072 (packed)		
16	384	8.5%
32	768	4.2%
64	1536	4.9%
Input Size: 5376x1024x5376 (packed)		
32	768	5.3%
Input Size: 5376x1024x5376 (underpopulated)		
32	384	4.1%
64	768	9.9%



Future Work

FUJITSU

- Devise general purpose strategies for overlapping communication and computation for use in mathematical libraries.
- With availability of non-blocking collectives widely expected in MPI-3 consider auto-tuning strategies for tuning MPI_Test() calls or make algorithms resistant to this.
- Exploit hyper-dimensionality in seemingly 3D algorithms to maximize utilization of hyper-dimensional network architectures [2] as they become common. Non-blocking collectives will be important here.

Acknowledgements

–Dr Ross Nobes

–Dr Nicholas Wilson

–Professor Daisuke Takahashi, Tsukuba University

–Dr Ning Li and Dr Mark Richardson at Numerical Algorithms Group Ltd

–Dr Stephen Pickles at STFC Daresbury Laboratory, UK

–Fujitsu Laboratories of Europe, UK

References

[1] Implementation and Performance Analysis of Non-Blocking Collective Operations for MPI, T. Hoefler, A. Lumsdaine and E. Rehm, http://www.unixer.de/publications/img/hoefler-sc07.pdf

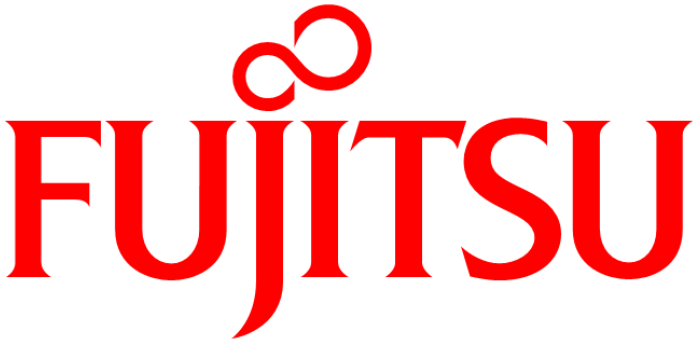
[2] Tofu: A 6D Mesh/Torus Interconnect for Exascale Computers, Y. Ajima, S. Sumimoto and T. Shimizu, Computer, 42(11), pp. 36–40 (2009).

[3] Overlapping Methods of All-to-All Communication and FFT Algorithms for Torus-connected Massively Parallel Supercomputers, J. Doi and Y. Negishi, SC10 (2010).

[4] An implementation of parallel 3-D FFT with 2-D Decomposition on a Massively Parallel Cluster of Multi-core Processors, D. Takahashi, PPAM 2009, Part 1, LNCS 6067, pp. 606-614 (2010).

[5] High performance and scalable non-blocking all-to-all with collective offload on Infiniband clusters: a study with parallel 3D FFT, Kandalla *et al*, Computer Science – Research and Development, **26**(3-4), 237-246 (2011).

[6] A Hybrid MPI/OpenMP Implementation of a Parallel 3-D FFT on SMP Clusters, D. Takahashi, PPAM 2005, LNCS 3911, pp. 970-977 (2006).



shaping tomorrow with you